

Vassili Harchenko

OpenWrt-tiedostopalvelin

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Tietoverkot

Insinöörityö

10.10.2015

Tekijä(t) Otsikko	Vassili Harchenko OpenWrt-tiedostopalvelin
Sivumäärä Aika	28 sivua 10.10.2015
Tutkinto	Insinööri (AMK)
Koulutusohjelma	Tietotekniikka
Suuntautumisvaihtoehto	Tietoverkot
Ohjaaja(t)	Koulutusohjelmavastaava Janne Salonen
<p>Opinnäytetyön aiheena oli perehtyminen OpenWrt-käyttöjärjestelmään reititinympäristössä ja sen mahdollistamaan palvelintoiminnallisuuteen Samba server -laajennuksen avulla. Työn tarkoituksena oli selvittää OpenWrt:n tuomat uudet ominaisuudet tietoverkkoja yhdistäville laitteille ja soveltaa yhtä niistä käytännössä. OpenWrt:n laitetuen kasvaessa ja käytön yleistyessä tämä selvitys oli ajankohtainen.</p> <p>Työ toteutettiin TP-Link TL-WDR4300 -reititinlaitteella ja VirtualBox-virtuaalikoneella, jota käytettiin OpenWrt-laiteohjelmiston kompilointiin ja asentamiseen sekä myöhemmin reitittimen palvelinominaisuuksien tutkimiseen. TL-WDR4300 valikoitui testilaitteeksi, koska se soveltui hyvin OpenWrt-käyttöön erinomaisen hinta-laatu-suhteensa vuoksi.</p> <p>OpenWrt-käyttöjärjestelmän kompilointi virtuaalikoneen Ubuntu-työympäristössä, ja sen asentaminen TL-WDR4300-reitittimeen todettiin onnistuneiksi. Reitittimeen oli tämän jälkeen asennettu tiedostopalvelinkäytön mahdollistavat laajennuspaketit ja ulkoinen USB-muisti. Kevyessä palvelinkäytössä OpenWrt-laiteohjelmisto ja sen laajennukset osoittautuivat selvityksen perusteella toimivaksi lisäksi reititinlaitteen muun toiminnallisuuden rinnalle.</p>	
Avainsanat	OpenWrt, Ubuntu, tiedostopalvelin, lähiverkko, VirtualBox, Samba

Author(s) Title	Vassili Harchenko OpenWrt file server
Number of Pages Date	28 pages 10 October 2015
Degree	Bachelor of Engineering
Degree Programme	Information Technology
Specialisation option	Data Networks
Instructor(s)	Janne Salonen, Head of Degree Programme
<p>The subject of thesis was to observe OpenWrt operating system in a routing environment as well as Samba file server functionality that it brings. The objective was to get familiar with the features provided by OpenWrt in terms of new functionality for routing devices, and to apply one of these new features in practice. As device support as well as the amount of users of OpenWrt grows constantly, this report is topical.</p> <p>The study was done with the use of TP-Link TL-WDR4300 router and a VirtualBox virtual machine, which was used for compiling and installing OpenWrt firmware, and later for testing newly installed server functionality of the router. TL-WDR4300 was chosen as a test device because it fits well for OpenWrt use due to its good quality-price ratio.</p> <p>Compiling of the OpenWrt firmware inside the virtual machines Ubuntu environment and installing the operating system on a TL-WDR4300 test device was a success. The required additional OpenWrt packages were installed on the router to enable file server functions, while external USB memory was also connected to the device. In moderately light file server use OpenWrt and its additional packages proved to be a working extension to routers overall functionality.</p>	
Keywords	OpenWrt, Ubuntu, file server, local area network, VirtualBox, Samba

Sisällys

Lyhenteet

1 Johdanto	1
2 OpenWrt	2
2.1 Historia	3
2.2 Kehitysympäristö	3
2.3 Yhteisö	4
2.4 GPL v2-lisenssi	5
3 VirtualBox	7
3.1 Virtuaalikone	8
3.1.2 Ubuntun asentaminen	8
3.1.2 Ubuntun käyttöönotto	11
3.2 Buildrootin rakentaminen	11
3.2.1 Laiteohjelmiston määrittäminen	12
3.2.2 Laiteohjelmiston kompilointi	13
4 OpenWrt-reititin	15
4.1 Laiteohjelmiston asennus reitittimeen	16
4.1.1 Yhteyden muodostaminen	19
4.1.2 Kirjautuminen sisään	19
4.2 Laiteohjelmiston yleiskatsastus	20
4.3 Paketinhallinta	21
4.4 Samba-tiedostopalvelin	23
4.4.1 Laajennuspaketit	23
4.4.2 Konfigurointi	24
5 Ubuntu client	26
Yhteenveto	26
Lähteet	27

Lyhenteet

SSH	Secure Shell. Protokolla salatun tiedonsiirtoyhteyden muodostamiseen mahdollisesti turvattoman verkon yli.
OPKG	Open Package Management. Sulautetuille järjestelmille tarkoitettu pienikokoinen pakettinhallintajärjestelmä.
GPL	General Public License. Vapaiden ohjelmistojen julkaisemiseen tehty lisenssi.
SVN	Apache Subversion. Versionhallintajärjestelmä hajautetun työskentelyn mahdollistamiseksi tietoverkoissa.
IRC	Internet Relay Chat. Pikaviestittelypalvelu reaaliaikaiseen kommunikointiin internetin välityksellä.
GNU	GNU's Not Unix. Free Software Foundationin kehittämä UNIX-yhteensopiva ohjelmistojärjestelmä.
IP	Internet Protocol. TCP/IP-mallin internetkerroksessa toimiva protokolla joka huolehtii tietoliikennepakettien siirtämisestä.

1 Johdanto

Tässä opinnäytetyössä perehdymme avoimen lähdekoodin OpenWrt-ohjelmiston käyttöön ja sen tuomiin uusiin mahdollisuuksiin reititinlaitteiden saralle. Työn tarkoituksena on asentaa reitittimeen OpenWrt-laiteohjelmisto ja soveltaa käytännössä yhtä sen lukuisista laajennuksista: tiedostopalvelintoiminnallisuutta hyödyntäen Samba-palvelinlaajennusta. Tavoitteena on tarjota selkeä tapa kyseessä olevan ohjelmiston ymmärtämiseen sekä antaa yksi konkreettinen malli sen käyttöön ja sovelluksiin. Työssä tutkitaan avoimen lähdekoodin laiteohjelmiston soveltamista käytännön näkökulmasta. Paljon huomiota annetaan myös laiteohjelmiston asentamiselle laitteeseen ja laitteen asetusten konfiguroinnille. Työn toteutuksessa sovelletaan virtualisointia ja työympäristönä käytetään Ubuntu Linuxia.

Ajatus opinnäytetyön aiheesta tuli tutustuttuani OpenWrt Buildroot -työkaluun ja samankaltaisiin PandoraBox- sekä DD-WRT-ohjelmistoihin työpaikallani Innorange Oy:ssa. Opinnäytetyössä käytetty laite on TP-Linkin valmistama TL-WDR4300 langaton reititin ja virtuaalikoneen käyttöjärjestelmänä toimi OpenWrt-kotisivun suositusten mukainen Linux-distribuutio: Ubuntu (64 bit). [1.]

2 OpenWrt

OpenWrt on Linux-pohjainen käyttöjärjestelmä sulautetuille laitteille ja erityisesti langattomille reitittimille pitäen sisällä tuen monille erilaisille järjestelmäpiirialustoille. Sen sijaan, että OpenWrt olisi tehty toteuttamaan vain yhtä tiettyä toiminnallisuutta, se antaa käyttäjilleen vapaasti muokattavan ja modulaarisen käyttöympäristön sekä tehokkaan pakethallintajärjestelmän laiteohjelmiston monipuolisen käytön tukemiseksi. OpenWrt tarjoaa viitekehysten uusien ja innovatiivisten käyttötarkoitusten rakentamiselle sekä ohjelmistokehittäjille että peruskäyttäjille. Tällä hetkellä OpenWrt:tä käytetään ympäri maailmaa muun muassa sekä langallisissa että langattomissa verkkoratkaisuissa, teollisissa ohjainjärjestelmissä, roboteissa, sensoriverkoissa sekä VoIP-järjestelmissä.

OpenWrt:n päämääränä on tukea kehitystä ja etsiä uusia, ennennäkemättömiä tapoja sulautettujen järjestelmien käyttöön. [2.]

OpenWrt-käyttöjärjestelmä pohjautuu Linux-ytimeen ja sen pääkomponentteja ovat kernelin lisäksi util-linux, julkaisusta riippuen joko uClibc tai musl ja BusyBox. Nämä käyttöjärjestelmän komponentit ovat tarkasti optimoitu ottaen huomioon monien sulautettujen järjestelmien laitteistorajoitukset. Ne ovat tarpeeksi pieniä mahtumaan esimerkiksi kotireitittimien rajoitettuun muistimäärään. [3.]

OpenWrt-käyttöjärjestelmän konfigurointi tapahtuu komentorivillä hyödyntäen joko Telnet- tai SSH-yhteyttä, tai käyttäen OpenWrt:n selainpohjaista Luci-käyttöliittymää. Nykyään OpenWrt:lle on saatavilla noin 3500 valinnaista laajennuspakettia, joista jokaisella voi tuoda käyttöjärjestelmään lisää ominaisuuksia ja käyttötapoja. Nämä paketit ovat asennettavissa OPKG-paketinhallintajärjestelmän kautta. [4.]

2.1 Historia

OpenWrt-projektin teki mahdolliseksi se, että verkkolaitevalmistaja Linksys kehitti WRT54G-reitittimelle laiteohjelmiston käyttäen siihen pohjana julkisesti saatavilla olevaa, GPL-lisenssin alaista avointa lähdekoodia. GPL-lisenssin ehtojen takia Linksysin oli julkaistava laiteohjelmistonsa lähdekoodi myös muiden käyttöön, kun asia pääsi julkisuuteen. Tämän tuloksena riippumattomat ohjelmistokehittäjät pääsivät tekemään tästä lähdekoodista omia, johdannaisia versioita. Käyttäen alkuperäistä koodia aluksi oman koodinsa pohjana ja myöhemmin referenssinä, OpenWrt-ohjelmistokehittäjät loivat Linux-distribuution, joka tarjosi sellaisia ominaisuuksia ja tapoja laitteiston hyödyntämiseen, joita ei aiemmin löytynyt kuluttajatasen reitittimistä. Ainoastaan WRT54G-laitesarja oli lähtökohtaisesti tuettu, mutta vuosien aikana tuki kasvoi kattamaan myös monia muita piirisarjoja ja laitetyppejä. Mukaan on mahtunut lukuisia isoja laitevalmistajia, kuten esimerkiksi TP-Link, ZyXel ja D-Link. [5.]

Erilaisten OpenWrt-kehityshaarojen nimet on annettu juomasekoitusten mukaan, ja ne pitävät sisällään seuraavat julkaisut: White Russian, Kamikaze, Backfire, Attitude Adjustment, Barrier Breaker ja Chaos Calmer. [6.]

Stable version	Release date	Naming	SVN
Chaos Calmer 15.05	2015 September	Chaos Calmer	r46764
Barrier Breaker 14.07	2014 October	Barrier Breaker	r42652
Attitude Adjustment 12.09	2013 April	Attitude Adjustment	r36088
Backfire 10.03.1	2011 December	Backfire	r29592
Backfire 10.03	2010 April	Backfire	r20728
Kamikaze 8.09.2	2010 January	Kamikaze	r18801
Kamikaze 8.09.1	2009 June	Kamikaze	r16278
Kamikaze 8.09	2008 September	Kamikaze	r14510
Kamikaze 7.09	2007 September	Kamikaze	r8679
Kamikaze 7.07	2007 July	Kamikaze	r7832
Kamikaze 7.06	2007 June	Kamikaze	r7204
White Russian 0.9	2007 January	White Russian	r6257

Taulukko 1. OpenWrt-julkaisuja. Mukana lueteltuina ovat myös eri versioiden revision-numerot SVN-versionhallintajärjestelmässä

2.2 Kehitysympäristö

OpenWrt:n kehitysympäristö ja kompiloitijärjestelmä tunnetaan yhdessä nimellä OpenWrt Buildroot, ja se perustuu vahvasti muokattuun Linuxin Buildroot-ohjelmistoon. OpenWrt Buildroot automatisoi OpenWrt-laiteohjelmiston kompiloinnin sulautetuille laitteille käyttäen cross-compilation toolchain-työkalua ja sisältää joukon päivityksiä ja Makefile-tiedostoja. Sen lisäksi, että OpenWrt Buildroot antaa käyttäjilleen työkalun laitekohtaiseen laiteohjelmiston kääntämiseen, se tarjoaa valmiiksi myös helppokäyttöisen ympäristön OpenWrt-laajennuspakettien tekoon. [7.]

Sulautetut laitteet kuten verkkoreitittimet yleensä sisältävät erilaisen prosessorin kuin mitä laiteohjelmiston kompiloinnin suorittava tietokone. Tämän takia osana OpenWrt Buildroot -ohjelmaa tarvitaan myös cross-compilation toolchain, työkalu, joka kääntää kohdelaitteen prosessoriarkkitehtuuriin sopivaa koodia. [7.]

Esimerkiksi jos isäntälaitte käyttää x86-prosessoriarkkitehtuuria ja kohdelaitte käyttää sulautetuissa järjestelmissä usein esiintyvää MIPS32-arkkitehtuuria. Tavallinen compilation toolchain generoisi koodia pelkästään x86-arkkitehtuurille, siinä missä cross-compilation toolchain kääntää koodin MIPS32-prosessoriarkkitehtuuriin sopivaksi. OpenWrt Buildroot automatisoi tämän käännösprosessin tukemaan lukuisten eri järjestelmien käskykanta-arkkitehtuureja. [7.]

OpenWrt Buildrootin keskeisimpiä ominaisuuksia ovat:

- laiteohjelmistojen automatisoitu kompilointi eri kohdealustoille
- Linux kernel menuconfigin käyttäminen asetusten konfigurointiin
- integroitujen cross-compiler toolchain-työkalujen tarjoaminen
- vastaaminen OpenWrt-laiteohjelmiston kompiloinnin työkalusta.

2.3 Yhteisö

OpenWrt-kehitystyö on monella tapaa läpinäkyvä prosessi, ja projektiin osallistujien yleisimpiä työkaluja ovat OpenWrt development center-internetsivu, IRC-kanavat sekä keskustelufoorumi. Kehitysprosessi perustuu paljolti aktiiviseen osallistumiseen, ja ne henkilöt, jotka eivät vielä ole virallisia OpenWrt-ohjelmistokehittäjiä ja ovat ilman kirjoitusoikeuksia OpenWrt:n repository-palvelimelle, voivat toimittaa tekemiään päivityksiä OpenWrt-järjestelmään joko sähköpostitse tai Trac-työkalun kautta. [8.]

Lähetetyt päivitykset joutuvat virallisten ohjelmistokehittäjien tarkasteluun, ja ne joko hyväksytään tai hylätään muun muassa päivitysten ohjelmakoodin laadun ja päivityksistä saatavan hyödyn perusteella. Jos päivitys joutuu hylätyksi, niin on hyvin todennäköistä saada se hyväksytyksi seuraavalla palautuskerralla korjattuaan mahdolliset epäkohdat.

Yhteistyö aktiivisen OpenWrt-kehittäjän kanssa IRC-kanavan välityksellä on monien kohdalla myös olennainen osa kehitystyötä. Mitä tulee viralliseksi ohjelmistokehittäjäksi pääsyyn, kirjoitusoikeuksiin repository-palvelimelle ja omien tietojensa saantiin viralliselle OpenWrt-kehittäjien sähköpostilistalle, niin osallistumalla päivitysten tekoon ja OpenWrt:n ohjelmistokehitystä koskeviin keskusteluihin käyttäjä pystyy saavuttamaan ohjelmistokehittäjätasoisien käyttäjätunnuksen OpenWrt development center-sivuille. [8.]

Suurin osa muutoksista OpenWrt-järjestelmään tapahtuu tällä hetkellä päivitysten muodossa ja yksi työkaluista niiden kirjoittamiseen on SVN eli Apache Subversion. Tämän versionhallintaohjelmiston tarkoitus on mahdollistaa muun muassa lähdekoodin muokkaamisen internetin välityksellä ja muokattavan materiaalin synkronoinnin. [9.]

OpenWrt:n tämän hetken virallisena versionhallinta-sovelluksena voidaan pitää Git-nimistä ohjelmistoa. Se on Apache Subversionia uudempi ja tukee paremmin laajasti hajautettua työskentelyä. Siinä käyttäjällä on muokattavan koodin pääsäilytyspaikan lisäksi oma paikallinen tietokanta, johon hän voi tehdä muutoksia jopa silloin, kun OpenWrt-päätielokantaan ei ole suoraa yhteyttä. Näin muutosten synkronointi onnistuu esimerkiksi siinä vaiheessa kun yhteys tietokantaan on taas olemassa. Tällä tavalla etätyöskentely projektin parissa on mahdollista jopa ilman internetyhteyttä. [9; 10.]

Git on myös suositeltava aloitteleville OpenWrt-kehittäjille, koska sen avulla on helppo kirjoittaa ja lähettää päivitystiedostoja esimerkiksi sähköpostina virallisille ohjelmistokehittäjille siinä vaiheessa, kun omia kirjoitusoikeuksia verkossa olevaan tietokantaan ei vielä ole.

2.4 GPL v2 -lisenssi

OpenWrt on ilmainen ohjelmisto, ja se tarjotaan sellaisenaan ilman takuusitoumusta. Ellei erikseen lähdetiedostoissa ole asiasta mainittu, OpenWrt ja sen kompiloitijärjestelmä tarjotaan GNU General Public License Version 2 –alaisuudessa. [11; 12.]

GPL v2 -lisenssi antaa käyttäjälle oikeuksia tietyillä ehdoilla:

- Käyttäjällä on oikeus kopioida ja levittää ohjelman muokkaamatonta lähdekoodia.
- Oikeus muokata ohjelman lähdekoodia ja levittää muokkaamansa koodia.
- Oikeus levittää kompiloituja ohjelmaversioita, sekä muokattuja että muokkaamattomia mikäli:
- Kaikki levitetyt kopiot sisältävät copyright-merkinnän ja merkinnän takuusitoutumattomuudesta.
- Kaikki muokatut kopiot levitetään GPL v2:n alaisina.

- Kaikkien kompiloitujen ohjelmaversioiden liitteenä on ajankohtaista lähdekoodia tai lupaus ajankohtaisen lähdekoodin tarjoamisesta

OpenWrt-distribuutio sisältää suuren määrän kolmannen osapuolen applikaatioita ja moduuleita, jotka ovat lukuisten avoimen lähdekoodin lisenssien alaisina. Lähdekoodi näihin ohjelmiin on saatavilla OpenWrt-toisiopalvelimelta, ja jokainen käyttäjä voi tarkistaa sitä kautta näiden applikaatioiden ja moduulien ajankohtaisen lisenssitilanteen. [12.]

GNU-projekti, joka lopulta mahdollisti GPL v2 -lisenssin alkoi vuonna 1984, kun Richard Stallman, ohjelmoija joka siihen aikaan teki töitä Massachusettsin Institute of Technologyssa, päätti luoda joukon ohjelmistoja joita voisi levittää ja soveltaa vapaasti ilman niiden käyttöä rajoittavaa linsensiointia. Stallman oli turhautunut sen ajan ohjelmistokulttuuriin, joka rajoitti ohjelmistojen käyttöä. Hän kehitti Hurd - käyttöjärjestelmän ja sen rinnalle GNU toolsetin, joka oli saatavilla vapaasti kaikille ilmaisella lisenssillä. Vuonna 1985 Richard Stallman perusti Free Software Foundationin eli FSF:n, voittoa tavoittelemattoman organisaation, jonka päämäärä oli tukea ilmaisia ohjelmistoprojekteja kuten GNU:ta ja myös luoda ohjelmistolisenssejä, jotka täyttäsivät hänen aloittamansa free software -liikkeen päämääriä. [13.]

Ennen vuotta 1989 FSF:n lisenssejä oli kirjoitettu ohjelmistokohtaisesti, mutta sinä vuonna ilmestynyt General Public License v1 muutti tämän viittaamalla ohjelmistoihin yleisesti, eikä niiden nimillä, joten tällaiseen lisenssiin pystyi liittämään suuren määrän vapaan levityksen ohjelmistoja. Vuonna 1991 GPL v1 oli päivitetty GPL v2 -muotoon ja yksi sen merkittävimpiä muutoksia oli niin sanottu Liberty or Death -ehto, joka oli lisätty korostamaan sitä, että GPL:n alaisia ohjelmistoja joko levitetään täysin GPL:n periaatteiden mukaisesti tai ei levitetä lainkaan. [13.]

3 VirtualBox

Tässä työssä OpenWrt-kompilointiin ja myöhemmin, OpenWrt-reitittimen palvelintoiminnallisuuden testaamiseen käytettävä Ubuntu 64 bit - tietokone rakennetaan virtuaaliympäristöön hyödyntäen VirtualBox-ohjelmistoa.

Oracle VM VirtualBox on tietokoneohjelmisto x86-virtualisointiin, jonka aikanaan oli kehittänyt Innotek GmbH -tietokoneohjelmistoyritys. VirtualBox ostettiin Innotek GmbH:lta Sun Microsystemsin toimesta ja nykyään sitä kehittää Oracle-korporaatio osana virtualisointituotteidensa joukkoa. [14.]

Tietojenkäsittelyssä x86-virtualisointi on keino mahdollistaa lukuisten käyttöjärjestelmien suorittamisen samalla x86-prosessorilla turvallisesti ja tehokkaasti. 1990-luvun jälkipuoliskolla virtualisointi saavutettiin monimutkaisten tietokoneohjelmistoratkaisujen avulla, koska sen ajan kuluttajatasen prosessoreista vielä puuttui natiivi tuki tähän. 2000-luvulla sekä Intel, että AMD lisäsivät prosessoreihinsa tuen virtualisoinnille mahdollistaen yksinkertaisemmat virtualisointiohjelmat ja myös ohjelmien tehokkuuden nousun. [15.]

Oracle VM VirtualBox toimii niin, että se asennetaan isäntäkoneen käyttöjärjestelmälle sovelluksena. Tämä sovellus sitten mahdollistaa uusien vieraskäyttöliittymien suorittamisen omassa virtuaaliympäristössä. Tuettuja käyttöjärjestelmiä ovat muun muassa Linux, Mac OS X, yleisimmät Windows-käyttöjärjestelmät, Solaris ja OpenSolaris. [14.]

Käytämme työympäristönä 64-bittistä Ubuntun 14.04.2-julkaisua OpenWrt-sivuston ohjeiden mukaisesti ja suoritamme Ubuntu-käyttöjärjestelmää VirtualBox-virtuaalikoneen sisällä. Linux-järjestelmä asennetaan näin virtuaalikoneeseen, jota suorittava VirtualBox-sovellus on samalla osana työssä käytettävän isäntäkoneen Windows-ympäristöä.

Virtualisoinnista on tämän työn kannalta lukuisia hyötyjä: se tarjoaa tehokkaan tavan kahden rinnakkaisen käyttöjärjestelmän samanaikaiseen käyttöön, helpon virtuaalikoneen käyttöjärjestelmän varmuuskopiointiin VirtualBoxin snapshot-toiminnolla ja ennen kaikkea mahdollisuuden soveltaa OpenWrt Buildroot työkalua, jonka käyttö ei ole tuettu Windowsissa. [1.]

3.1 Virtuaalikone

Koska VirtualBox suorittaa yhden käyttöjärjestelmän toimintaa toisen käyttöjärjestelmän sisällä ja hyödyntää samoja laitteistoresursseja, kannattaa pitää mielessä se, että esimerkiksi tietokoneen keskusmuistin määrän tulisi olla riittävä, jotta kummankin käyttöjärjestelmän toiminta olisi optimaalista. Moniytimisen prosessorin käyttö on myös suositeltavaa. VirtualBoxin tarjoaman virtualisoinnin erona esimerkiksi perinteiseen monissa tietokoneissa mahdolliseen kaksoiskäynnistykseen (dual boot), jossa käyttöjärjestelmät asennetaan suoraan omille kovalevyosioilleen on se, että kaksoiskäynnistys mahdollistaisi vain yhden käyttöjärjestelmän suorittamisen samalla kerralla. [16.]

Virtualisoidun käyttöjärjestelmän asennuksen koon ei tarvitse olla VirtualBox-ohjelmistossa välttämättä ennalta määritelty vaan se voi olla dynaamisesti kokoaan muuttava virtuaalilevy, jonka potentiaalista maksimikokoa rajoittaa lopulta vain tietokoneen todellisen kovalevyn tila. Vaihtaakseen käyttöjärjestelmien välillä käyttäjän ei tarvitse käynnistää tietokonetta uudelleen, vaan käyttöjärjestelmien vaihtaminen sekä niiden välinen tiedostojen siirto tapahtuu saumattomasti. [17.]

Virtuaalikone soveltaa jo vakioasetuksilla isäntäkoneen verkkokorttia ja sitä kautta internetyhteyttä, joten käyttäjän ei tarvitse huolehtia esimerkiksi oikeiden verkkokortin ajureiden etsimisestä Ubuntulle. Sama koskee grafiikkakortin ajureita ja VirtualBoxin guest additions-laajennuksen avulla Ubuntu-virtuaalikoneelle saadaan asetettua muun muassa skaalautuva näyttöresoluutio. [17.]

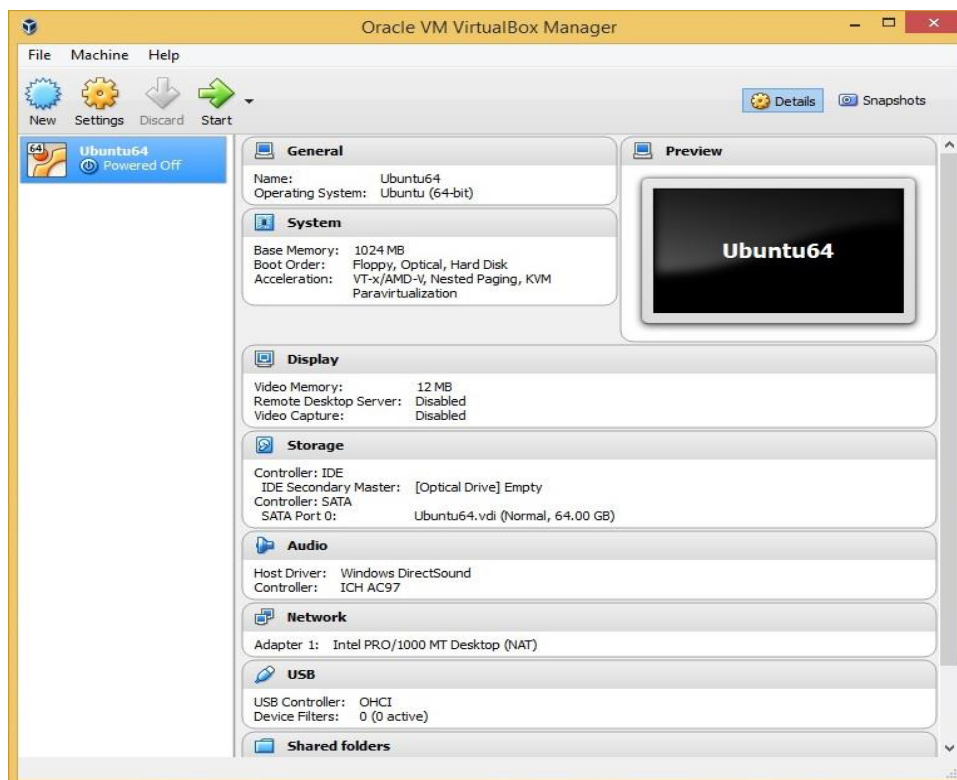
Tietokoneen laitteiston hyödyntämisen kannalta virtualisointia voidaan pitää dual boot -vaihtoehtoa vähemmän tehokkaana, koska vieraskoneen käyttöjärjestelmän suorittaminen isäntäkoneen käyttöjärjestelmän päällä joka tapauksessa tarkoittaa sitä, että tietokoneen laitteisto joutuu vastaamaan kahden käyttöjärjestelmän resurssitarpeista.

3.1.1 Ubuntun asentaminen

Ladattuamme VirtualBoxin Oracle VM VirtualBox -kotisivulta, suoritamme sovelluksen asentamisen muiden Windows-ohjelmien asennuksen tapaan. Tämän jälkeen haemme Ubuntu 64-bit levykuvan Ubuntun internetsivuilta ISO-tiedostomuodossa

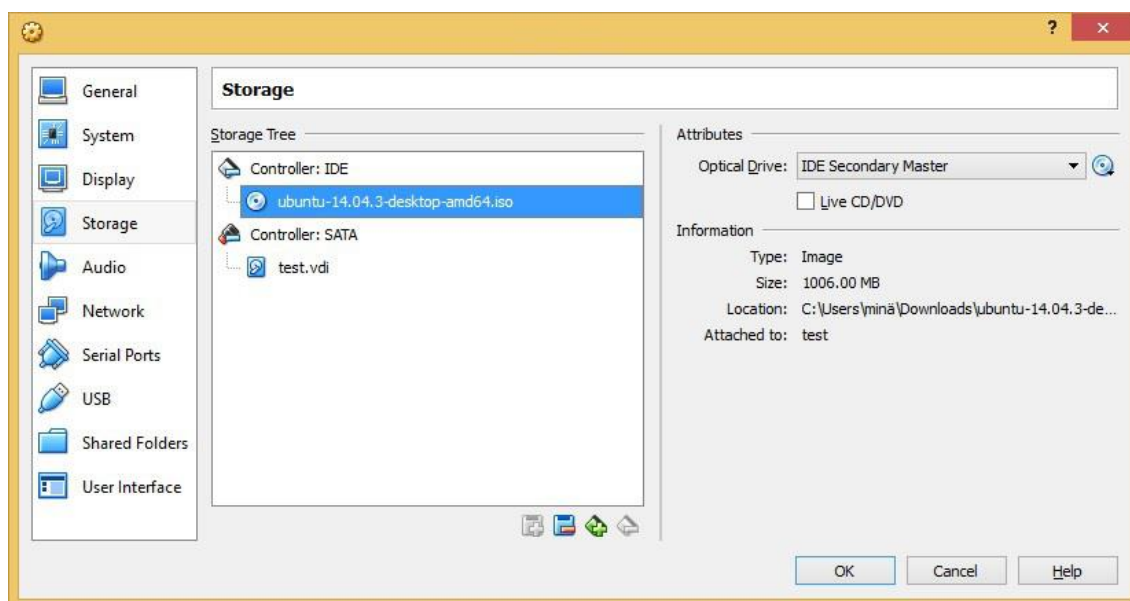
tietokoneellemme. Kun olemme käynnistäneet VirtualBoxin Windows-startup menun kautta, valitsemme New'n luodaksemme uuden virtuaalikoneen New Virtual Machine Wizardin avulla. Annamme virtuaalikoneellemme nimeksi Ubuntu64 ja spesifioimme käyttöjärjestelmän olevan Ubuntu Linux.

VirtualBox yrittää aluksi itse päätellä virtuaalikoneelle jaettavan keskusmuistin määrän, mutta koska meillä on käytössä 4 GB keskusmuistia, valitsemme määräksi 1 GB tarjoamaan riittävästi keskusmuistia muun muassa tulevaan laiteohjelmiston kompiloitiprosessiin. Tämän jälkeen luomme wizardissa virtuaalikoneelle dynaamisesti laajenevan virtuaalilevyn, koska se tulee skaalautumaan virtuaalikoneen käyttämän tilan mukaan ja on siten meidän kannalta suoraviivaisempi vaihtoehto kuin kiinteä kokorajoitus.



Kuva 1. Virtuaalikone ja sen järjestelmätiedot VirtualBox-ympäristössä

Seuraavaksi lisäämme aiemmin ladatun Ubuntu levykuvan suoritettavaksi virtuaalikoneen käynnistymisen yhteydessä. Tämä tapahtuu settings storage -valikon kautta, jonka CD/DVD Device -osiossa me valitsemme Ubuntu Linux ISO -tiedoston.



Kuva 2. Ubuntun levykuva virtuaalikoneen emuloidussa CD/DVD-asemassa

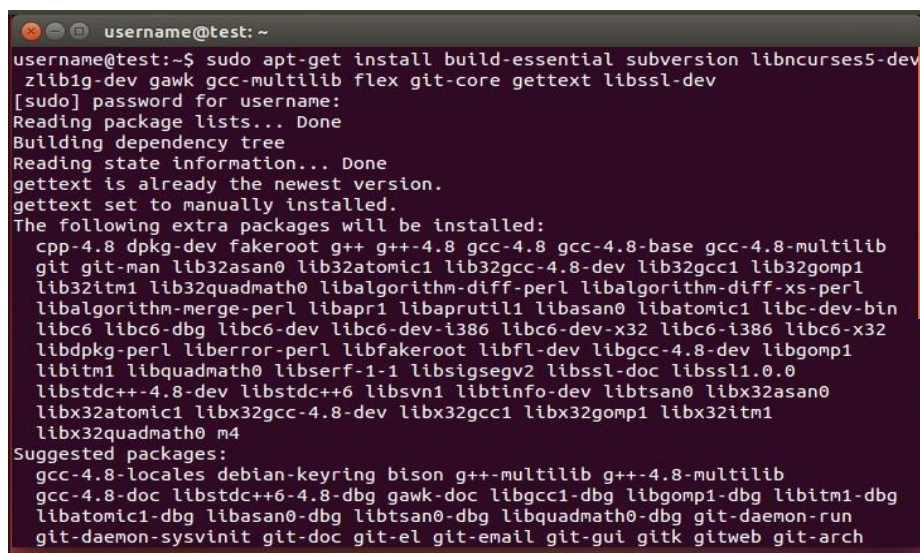
Kun olemme käynnistäneet virtuaalikoneen, myös Ubuntun virtuaalinen asennuslevy enabloituu, ja käymme läpi tavalliset asennusvaiheet. Valittuamme kielen eteemme tulee Install Options –valikko, josta aloitamme Ubuntun asennuksen virtuaalikoneeseen. Seuraavaksi Ubuntu tarjoaa meille vaihtoehtona käyttöjärjestelmän päivittämisen ajan tasalle internetin kautta asennusprosessin aikana, ja valitsemme myös Ubuntun asentamisen virtuaalilevyille, aikavyöhykkeen, sekä default keyboard layoutin. Kirjoitamme käyttäjänimeksi test, ja tämä automaattisesti täydentää puolestamme myös login name -kohdan sekä tietokoneen nimen. Salasanaksi asetamme Qwerty123. Päästyämme asennusprosessin loppuun, valitsemme restart now'n sen viimeistelemiseksi.

Virtuaalikoneen uudelleenkäynnistymisen jälkeen Ubuntu-käyttöjärjestelmä latautuu, ja kirjaudumme sisään käyttäjänimeä ja salasanaa käyttäen. Valitsemme virtuaalikoneen kohdalta devices ja Insert Guest Additions CD Image. Ubuntun kysyttyä salasanaa tämän VirtualBoxin lisäsovelluksen asennusta varten valitsemme install now. Kun olemme antaneet ohjelman suoriutua, käynnistämme virtuaalikoneen vielä kerran uudelleen. Virtuaalikoneen ikkunan resoluutiota voi nyt vaihtaa vapaasti, ja tämä uusi ominaisuus löytyy View Auto-resize Guest Display -valikosta pysyvästi.

3.1.2 Ubuntun käyttöönotto

Kun olemme kirjautuneet sisään, avaa Ubuntu Terminal-komentorivityökalun. Sen avulla navigoimme työpöydälle `cd Desktop`-komennolla. Luomme työpöydälle nyt hakemiston tulevalle OpenWrt Buildroot asennukselle komennolla `mkdir testFolder`. Ennen kuin aloitamme OpenWrt Buildrootin rakentamisen lataamme Ubuntuille tarvittavat työkalut tulevaa OpenWrt-laiteohjelmiston kompiloitua varten.

Päivitämme aluksi Ubuntun ajan tasalle komennolla `sudo apt-get update`. Sitten asennamme git-työkalun OpenWrt Buildrootin lataamiseksi ja rakennustyökalut cross-compilation käännösprosessiin seuraavilla komennoilla: `sudo apt-get install git-core build-essential libssl-dev libncurses5-dev unzip` ja `sudo apt-get install build-essential subversion libncurses5-dev zlib1g-dev gawk gcc-multilib flex git-core gettext libssl-dev`. [1.]

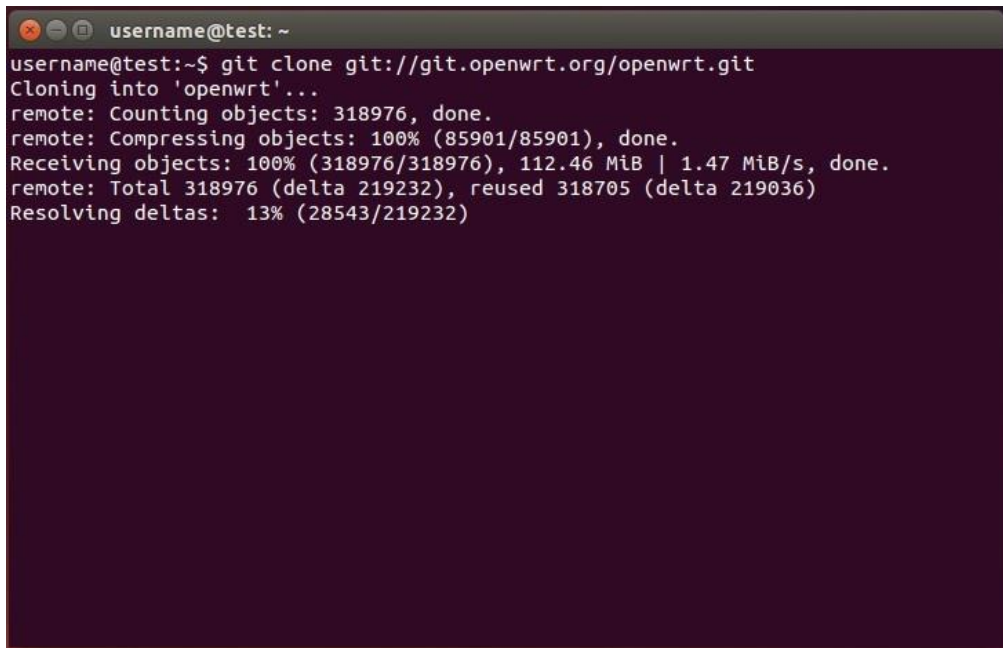


```
username@test: ~
username@test:~$ sudo apt-get install build-essential subversion libncurses5-dev
zlib1g-dev gawk gcc-multilib flex git-core gettext libssl-dev
[sudo] password for username:
Reading package lists... Done
Building dependency tree
Reading state information... Done
gettext is already the newest version.
gettext set to manually installed.
The following extra packages will be installed:
  cpp-4.8 dpkg-dev fakeroot g++ g++-4.8 gcc-4.8 gcc-4.8-base gcc-4.8-multilib
  git git-man lib32asan0 lib32atomic1 lib32gcc-4.8-dev lib32gcc1 lib32gomp1
  lib32itm1 lib32quadmath0 libalgorithm-diff-perl libalgorithm-diff-xs-perl
  libalgorithm-merge-perl libapr1 libaprutil1 libasan0 libatomic1 libc-dev-bin
  libc6 libc6-dbg libc6-dev libc6-dev-i386 libc6-dev-x32 libc6-i386 libc6-x32
  libdpkg-perl liberror-perl libfakeroot libfl-dev libgcc-4.8-dev libgomp1
  libitm1 libquadmath0 libserf-1-1 libsigsegv2 libssl-doc libssl1.0.0
  libstdc++-4.8-dev libstdc++6 libsvn1 libtinfo-dev libtsan0 libx32asan0
  libx32atomic1 libx32gcc-4.8-dev libx32gcc1 libx32gomp1 libx32itm1
  libx32quadmath0 m4
Suggested packages:
  gcc-4.8-locales debian-keyring bison g++-multilib g++-4.8-multilib
  gcc-4.8-doc libstdc++6-4.8-dbg gawk-doc libgcc1-dbg libgomp1-dbg libitm1-dbg
  libatomic1-dbg libasan0-dbg libtsan0-dbg libquadmath0-dbg git-daemon-run
  git-daemon-sysvinit git-doc git-el git-email git-gui gitk gitweb git-arch
```

Kuva 3. Ubuntun terminaalin käyttö uusien sovellusten asentamiseen

3.2 Buildrootin rakentaminen

Siirrymme testFolder-hakemistoon `cd testFolder` -komennolla. Nyt voimme ladata uusimman saatavilla olevan OpenWrt Buildrootin käyttäen git-työkalua komennolla: `git clone git://git.openwrt.org/openwrt.git`. Tämä luo openwrt-nimisen alihakemiston testFolder-hakemistoon.

A terminal window with a dark background and light text. The prompt is 'username@test: ~'. The command entered is 'git clone git://git.openwrt.org/openwrt.git'. The output shows the cloning progress: 'Cloning into 'openwrt'...', 'remote: Counting objects: 318976, done.', 'remote: Compressing objects: 100% (85901/85901), done.', 'Receiving objects: 100% (318976/318976), 112.46 MiB | 1.47 MiB/s, done.', 'remote: Total 318976 (delta 219232), reused 318705 (delta 219036)', and 'Resolving deltas: 13% (28543/219232)'.

```
username@test: ~
username@test:~$ git clone git://git.openwrt.org/openwrt.git
Cloning into 'openwrt'...
remote: Counting objects: 318976, done.
remote: Compressing objects: 100% (85901/85901), done.
Receiving objects: 100% (318976/318976), 112.46 MiB | 1.47 MiB/s, done.
remote: Total 318976 (delta 219232), reused 318705 (delta 219036)
Resolving deltas: 13% (28543/219232)
```

Kuva 4. OpenWrt Buildrootin lataaminen Git-työkalulla

Openwrt-hakemiston sisällä on neljä olennaista alihakemistoa: tools, toolchain, package ja target. Tools ja toolchain viittaavat yleisiin työkaluihin, joita käytetään laiteohjelmiston rakentamiseen soveltaen compileria ja C-kirjastoa. Package-alihakemisto tulee sisältämään ipk-tiedostomuodon paketteja, joista suuri osa OpenWrt-laiteohjelmistossa esiintyvistä sovelluksista koostuu. Näiden ipk-pakettien avulla OpenWrt:n toiminnallisuutta pystytään muuttamaan ja laajentamaan riippuen siitä, mitä paketteja laiteohjelmistoon on käännetty mukaan. Laajennuspaketteja pystyy myös hakemaan valmiiseen OpenWrt käyttöjärjestelmään asennettavaksi muun muassa OpenWrt:n repository-palvelimelta jälkikäteen.

Target-alihakemisto viittaa kohdelaitteeseen, sulautetun järjestelmän alustaan, jolle firmwarea ollaan kääntämässä. Erityisen maininnan ansaitsee tämän hakemiston sisältämä Linux-alihakemisto, joka sisältää muun muassa kernelin päivityksiä ja eri alustojen konfiguraatioita sisältäviä alihakemistoja. [18.]

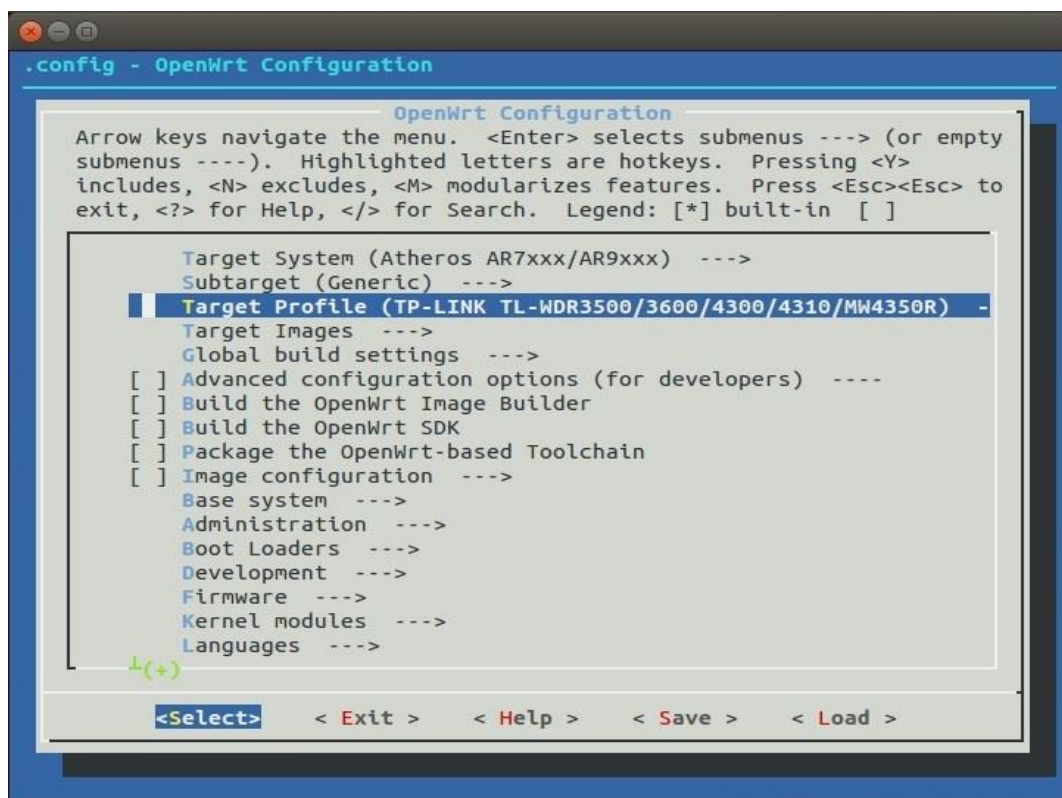
3.2.1 Laiteohjelmiston määrittäminen

Lataamme ja asennamme saatavilla olevat feeds-konfiguraatiot mahdollistamaan lukuisten ajantasalla olevien OpenWrt laajennuspakettien kääntämisen. Laskeudumme openwrt-hakemistoon `cd openwrt` -komennolla ja suoritamme seuraavat komennot:

`./scripts/feeds update -a, ./scripts/feeds install -a`. Tämän jälkeen kirjoitamme vielä komennot: `make defconfig`, `make prereq` ja `make menuconfig`. [1.]

`Make menuconfig` -komento aktivoi OpenWrt:n configuration menu screen -valikon. Tämän valikon kautta pystymme valitsemaan mille alustalle olemme kääntämässä laiteohjelmistoa, mitä toolchainin versiota haluamme käyttää kompilointiin ja mitä laajennuspaketteja haluamme asentaa laiteohjelmistoon jo käännösvaiheessa.

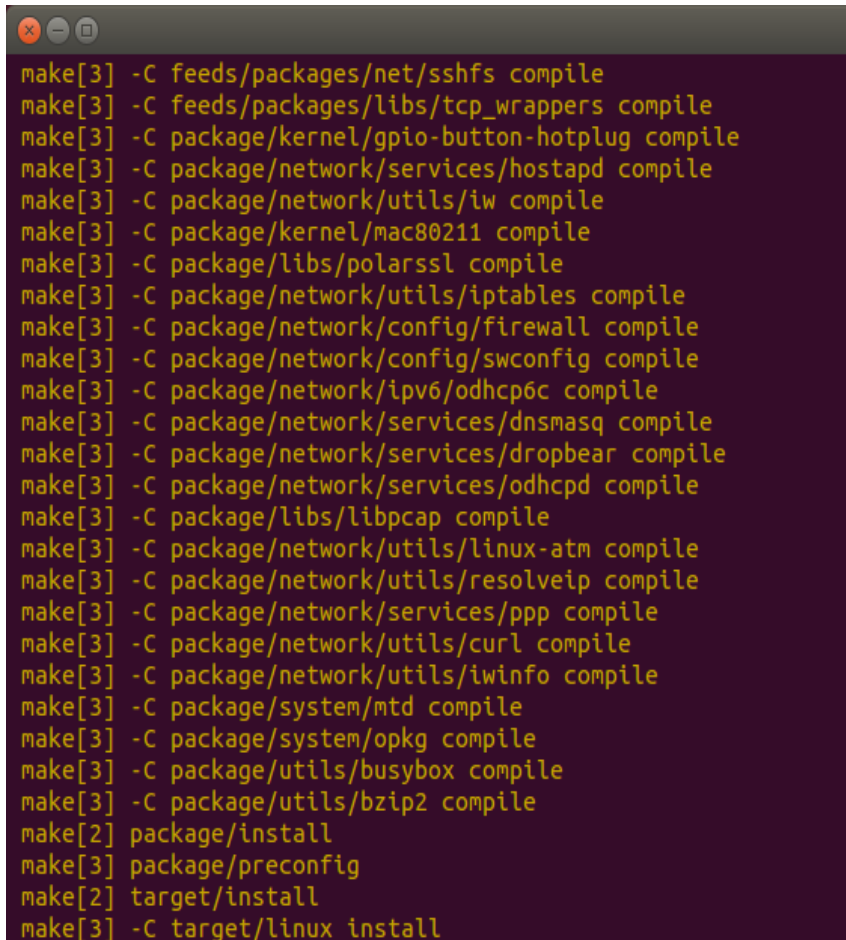
Valitsemme kompiloinnin kohdejärjestelmäksi Atheros AR7xxx/AR9xxx, koska siihen sisältyy myös WDR4300-reitittimen käyttämä system on a chip -alusta. Kohdeprofiiliksi valitsemme TP-LINK TL-WDR3500/3600/4300/4310/MW4350R. Tämän jälkeen valitsemme `exit` ja `save`.



Kuva 5. Kohdealustan määrittäminen kompiloitavalle laiteohjelmistolle

3.2.2 Laiteohjelmiston kompilointi

Firmwaren kompiloinnin aloittamiseksi kirjoitamme komentoriville `make`. Vakioasetuksilla OpenWrt kertoo käyttäjälle vain olennaisen tiedon kompiloinnin edistymisestä.

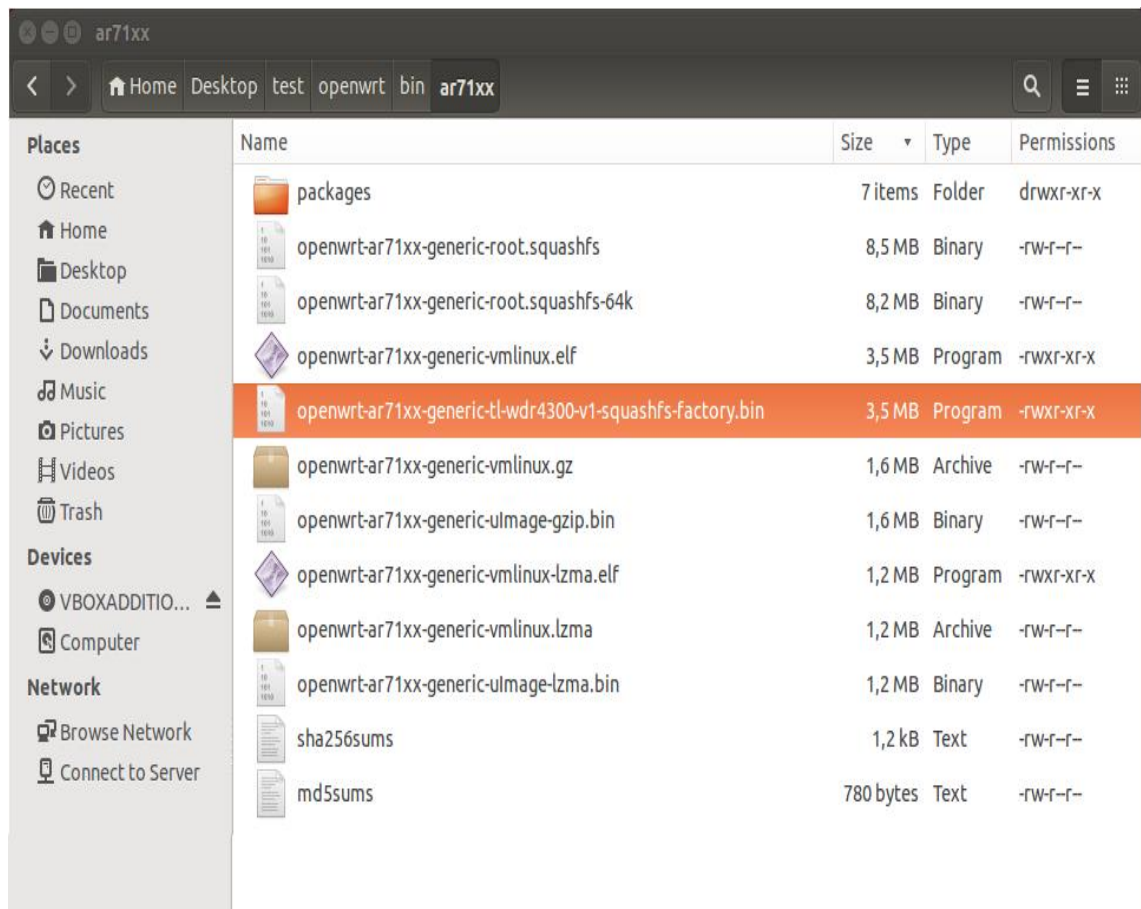


```
make[3] -C feeds/packages/net/sshfs compile
make[3] -C feeds/packages/libs/tcp_wrappers compile
make[3] -C package/kernel/gpio-button-hotplug compile
make[3] -C package/network/services/hostapd compile
make[3] -C package/network/utils/iw compile
make[3] -C package/kernel/mac80211 compile
make[3] -C package/libs/polarssl compile
make[3] -C package/network/utils/iptables compile
make[3] -C package/network/config/firewall compile
make[3] -C package/network/config/swconfig compile
make[3] -C package/network/ipv6/odhcp6c compile
make[3] -C package/network/services/dnsmasq compile
make[3] -C package/network/services/dropbear compile
make[3] -C package/network/services/odhcpd compile
make[3] -C package/libs/libpcap compile
make[3] -C package/network/utils/linux-atm compile
make[3] -C package/network/utils/resolveip compile
make[3] -C package/network/services/ppp compile
make[3] -C package/network/utils/curl compile
make[3] -C package/network/utils/iwinfo compile
make[3] -C package/system/mtd compile
make[3] -C package/system/opkg compile
make[3] -C package/utils/busybox compile
make[3] -C package/utils/bzip2 compile
make[2] package/install
make[3] package/preconfig
make[2] target/install
make[3] -C target/linux install
```

Kuva 6. OpenWrt Buildroot tarjoaa tietoa kompilointiprosessin edistymisestä reaaliaikaisesti

Tämä helpottaa edistymisen seuranta, mutta mikäli tarvitsemme perusteellisempaa tietoa esimerkiksi siihen, miksi kompilointi kaatuu johonkin tiettyyn virheeseen, on hyvä käyttää make V=99 komentoa, sillä se antaa paljon syvällisempää informaatiota kompilointiprosessista. [18.]

Kompilointiprosessin aikana OpenWrt Buildroot lataa lähdetiedostoja dl-hakemistoon ja aloittaa päivittämisen ja kääntämisen build_dir-hakemistoon. Kun kompilointi on päättynyt, valmis laiteohjelmisto sijoittuu automaattisesti bin-hakemistoon.



Kuva 7. Valmis OpenWrt-laiteohjelmisto laitekohtaisessa alihakemistossa ar71xx

4 OpenWrt-reititin

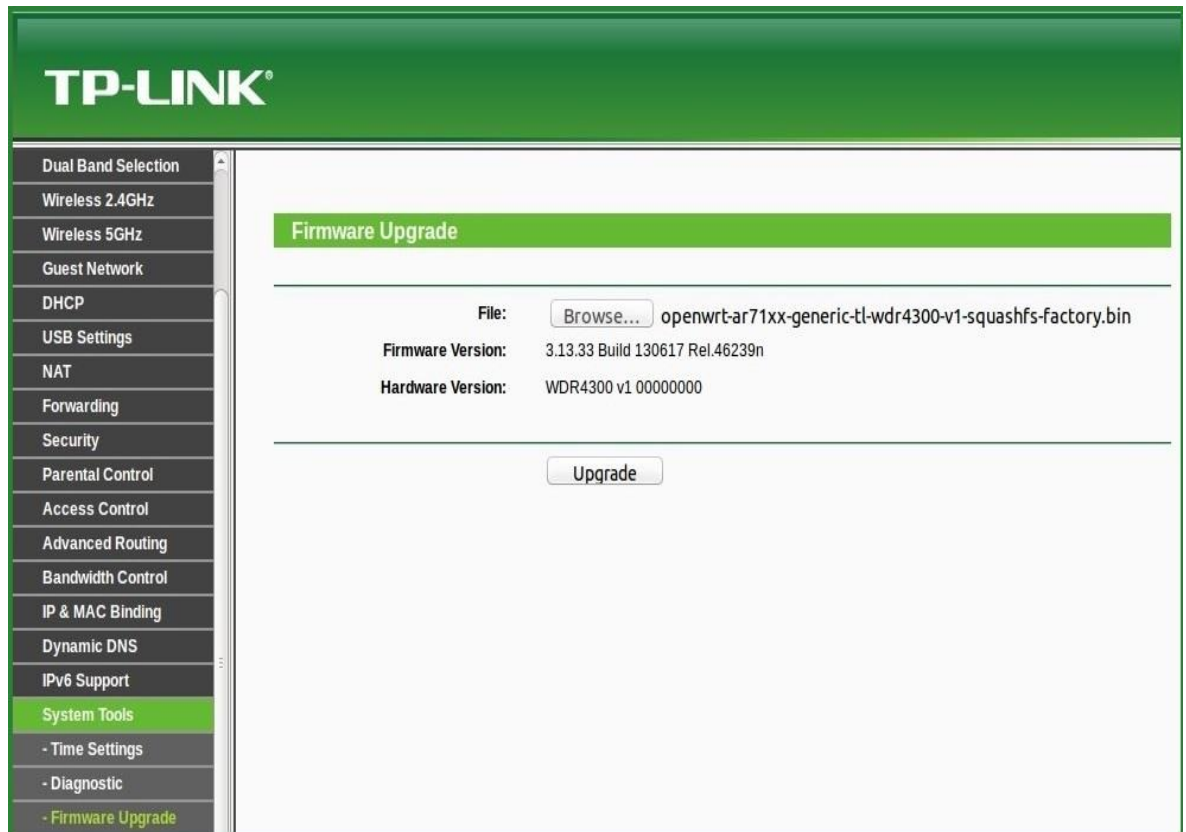
TP-Link TL-WDR4300 on dual band -reititin, joka tarjoaa sekä 2,4 GHz:n että 5 GHz:n Wi-Fi-verkkoa. Se sisältää neljä Gigabitin Ethernet porttia ja kaksi USB 2.0-porttia. Hintaansa nähden WDR4300 antaa laitteiston puolesta hyvän alustan OpenWrt-asennukselle, sekä kuuluu jo valmiiksi tuettujen OpenWrt-laitteiden joukkoon. Reititin sisältää myös 8 MB flash-muistia, 128 MB RAM-muistia sekä Atheros AR9344 -mallia olevan prosessorin. [19.]



Kuva 8. TP-Link TL-WDR4300 -reititin ja USB-lisämuisti

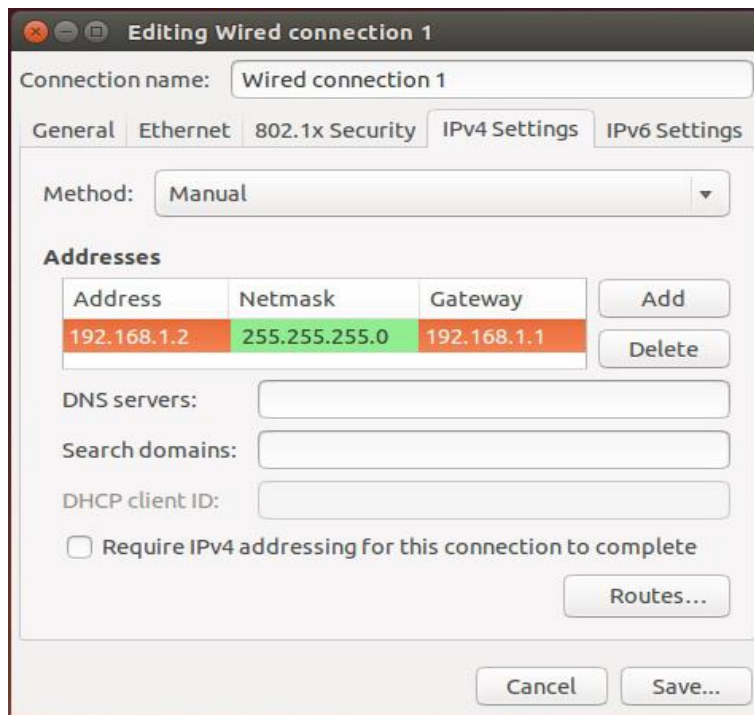
4.1 Laiteohjelmiston asennus reitittimeen

Asetamme virtuaalikoneen eth0-yhteyden IP-osoitteeksi 192.168.0.2, netmaskiksi 255.255.255.0 ja gatewayksi 192.168.0.1, eli kohdoreitittimen IP-osoitteen. Navigoimme selaimessa osoitteeseen 192.168.0.1 ja käytämme siellä admin-käyttäjätunnusta ja admin-salasanaa sisäänkirjautumiseen. Reitittimen omasta selainpohjaisesta käyttöliittymästä valitsemme Select System Tools ja sieltä Firmware Upgrade. Browsersen kautta navigoimme OpenWrt Buildrootin-bin hakemistoon ja valitsemme sieltä kääntämämme laiteohjelmiston.

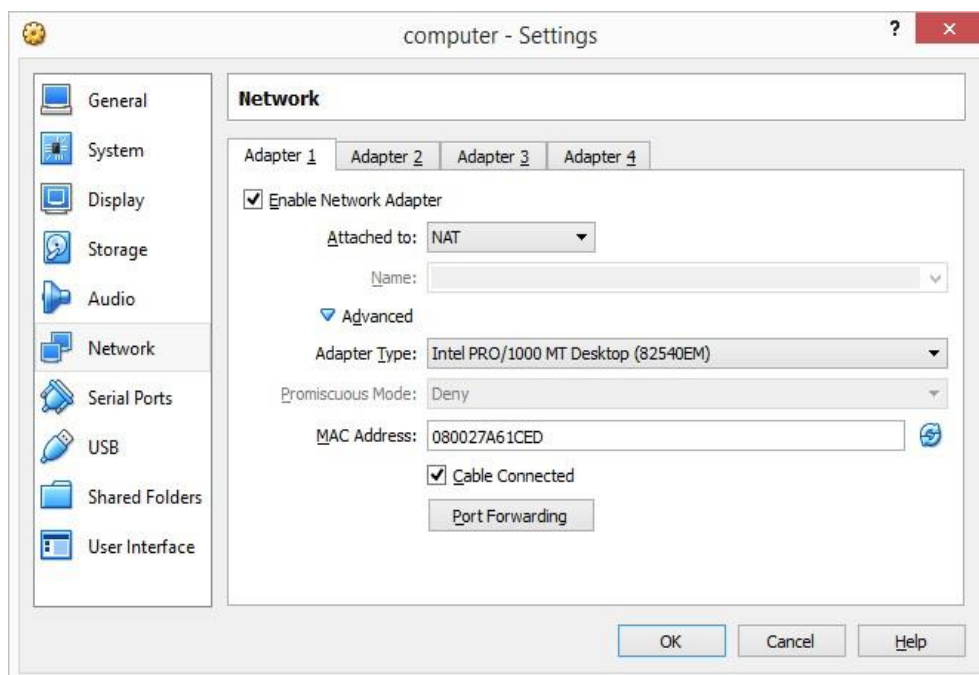


Kuva 9. Reitittimen alkuperäisen TP-Link laiteohjelmiston Firmware Upgrade -valikko

Valitsemalla tämän jälkeen Upgrade, alkaa OpenWrt-firmwaren asennus reititinlaitteeseen. Muutaman minuutin kuluttua ilmoitus vaihtuu muotoon Software Upgraded Successfully, ja reititinlaite käynnistyy uudelleen. Reitittimen verkko-osoite vaihtuu nyt TP-Linkin 192.168.0.1-default-osoitteesta, OpenWrt-firmwaren default IP -osoitteeksi, eli 192.168.1.1 muotoon laiteohjelmiston vaihtumisen yhteydessä.



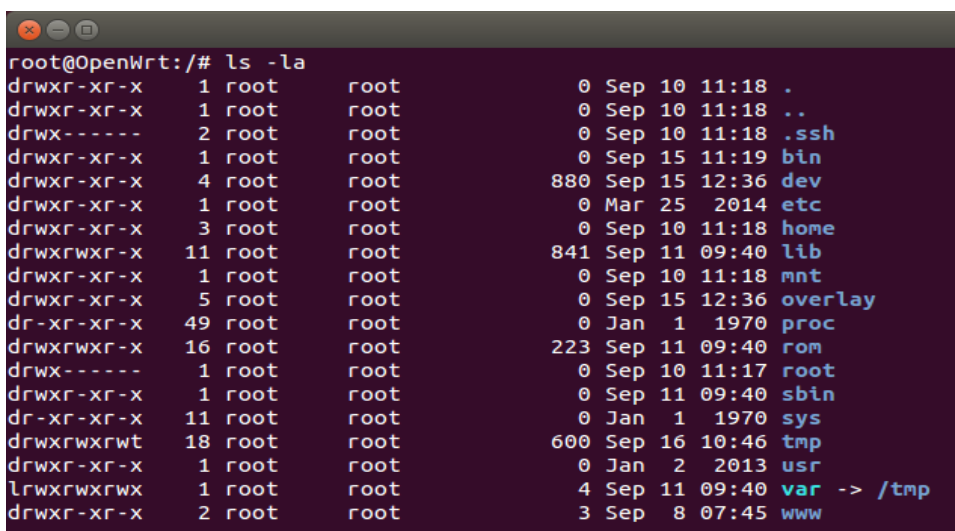
Kuva 10. Virtuaalikoneen uusien verkkoasetusten asettaminen yhteyden muodostamiseksi OpenWrt-firmware-reitittimeen



Kuva 11. Virtuaalikone hyödyntämässä isäntäkoneen verkkosovitinta

4.2 Laiteohjelmiston yleiskatsaus

Sisäänkirjautuneina voimme suorittaa `ls -la`-komennon. Tässä tapauksessa se tulostaa komentoriville listan käyttöjärjestelmän toiminnan kannalta olennaisista päähakemistoista.



```

root@OpenWrt:/# ls -la
drwxr-xr-x 1 root root      0 Sep 10 11:18 .
drwxr-xr-x 1 root root      0 Sep 10 11:18 ..
drwx----- 2 root root      0 Sep 10 11:18 .ssh
drwxr-xr-x 1 root root      0 Sep 15 11:19 bin
drwxr-xr-x 4 root root 880 Sep 15 12:36 dev
drwxr-xr-x 1 root root      0 Mar 25 2014 etc
drwxr-xr-x 3 root root      0 Sep 10 11:18 home
drwxrwxr-x 11 root root 841 Sep 11 09:40 lib
drwxr-xr-x 1 root root      0 Sep 10 11:18 mnt
drwxr-xr-x 5 root root      0 Sep 15 12:36 overlay
dr-xr-xr-x 49 root root      0 Jan  1 1970 proc
drwxrwxr-x 16 root root 223 Sep 11 09:40 rom
drwx----- 1 root root      0 Sep 10 11:17 root
drwxr-xr-x 1 root root      0 Sep 11 09:40/sbin
dr-xr-xr-x 11 root root      0 Jan  1 1970 sys
drwxrwxrwt 18 root root 600 Sep 16 10:46 tmp
drwxr-xr-x 1 root root      0 Jan  2 2013 usr
lrwxrwxrwx 1 root root      4 Sep 11 09:40 var -> /tmp
drwxr-xr-x 2 root root      3 Sep  8 07:45 www

```

Kuva 13. Komennon `ls` käyttöä OpenWrt-käyttöjärjestelmässä

Tämän työn kannalta tärkeitä hakemistoja ovat `etc` ja `mnt`. `Mnt` on hakemisto, johon tulemme käyttöjärjestelmän tasolla yhdistämään ulkoisen USB-lisämuistin. Komennolla `ls etc` voimme tarkastella, mitä alihakemistoja `etc` sisältää navigoimatta `cd`-komennolla hakemiston sisään.

`Etc` sisältää sellaisen meille tärkeän alihakemiston kuten `config`. `ls /etc/config` komennolla pääsemme näkemään, että se sisältää muun muassa `firewall`- ja `network`-konfiguraatiot. Ne määrittelevät palomuurin toiminnan ja sen, miten reitittimen verkkoasetukset on konfiguroitu. Networkin alta löytyvät muun muassa `lan`- ja `wan`-interface-asetukset.

```

# Copyright (C) 2006 OpenWrt.org

config interface loopback
    option ifname    lo
    option proto     static
    option ipaddr    127.0.0.1
    option netmask   255.0.0.0

config interface lan
    option ifname    eth0
    option type      bridge
    option proto     static
    option ipaddr    192.168.1.1
    option netmask   255.255.255.0

config interface wan
    option ifname    eth3
    option proto     dhcp
~
~
~
- network 1/21 4%

```

Kuva 14. OpenWrt-reitittimen verkkoasetukset

4.3 Paketinhallinta

Jotta voisimme laajentaa meidän nykyisen OpenWrt-käyttöjärjestelmän toiminallisuutta Samba-tiedostopalvelimeksi, tulemme tarvitsemaan ipk-tiedostomuodon laajennuspaketteja. Kaksi yleistä tapaa näiden pakettien tuottamiseksi on joko kompiloimalla ne laiteohjelmistoon mukaan jo valmiiksi soveltaen OpenWrt Buildroot -työkalua tai hyödyntämällä OpenWrt -repository palvelinta. Valitsimme jälkimmäisen vaihtoehdon, jotta voisimme paremmin demonstroida OPKG-paketinhallintajärjestelmää ja sen toimintaa.

Etc/opkg.conf- tiedostosta löytyvät url-polut, joista OPKG-paketinhallinta hakee listan saatavilla olevista laajennuspaketeista. Meidän kohdalla <http://downloads.openwrt-> url-polut on jo määritelty valmiiksi johtamaan openwrt-repositoryyn, mutta ne voi myös vaihtaa ja määrittää viittaaman tarvitsemaansa pakettivarastoon.

Työn kannalta OPKG:n default-konfiguraatio riittää, ja haemme uusimman listan saatavilla olevista laajennuspaketeista komennolla `opkg update`.

```

username@test: ~
root@OpenWrt:~# opkg update
Downloading http://downloads.openwrt.org/snapshots/trunk/ar71xx/generic/packages
/base/Packages.gz.
Updated list of available packages in /var/opkg-lists/chaos_calmer_base.
Downloading http://downloads.openwrt.org/snapshots/trunk/ar71xx/generic/packages
/base/Packages.sig.
Signature check passed.
Downloading http://downloads.openwrt.org/snapshots/trunk/ar71xx/generic/packages
/luci/Packages.gz.
Updated list of available packages in /var/opkg-lists/chaos_calmer_luci.
Downloading http://downloads.openwrt.org/snapshots/trunk/ar71xx/generic/packages
/luci/Packages.sig.
Signature check passed.
Downloading http://downloads.openwrt.org/snapshots/trunk/ar71xx/generic/packages
/management/Packages.gz.
Updated list of available packages in /var/opkg-lists/chaos_calmer_management.
Downloading http://downloads.openwrt.org/snapshots/trunk/ar71xx/generic/packages
/management/Packages.sig.
Signature check passed.
Downloading http://downloads.openwrt.org/snapshots/trunk/ar71xx/generic/packages
/packages/Packages.gz.
Updated list of available packages in /var/opkg-lists/chaos_calmer_packages.
Downloading http://downloads.openwrt.org/snapshots/trunk/ar71xx/generic/packages
/packages/Packages.sig.
Signature check passed.
Downloading http://downloads.openwrt.org/snapshots/trunk/ar71xx/generic/packages
/routing/Packages.gz.
Updated list of available packages in /var/opkg-lists/chaos_calmer_routing.
Downloading http://downloads.openwrt.org/snapshots/trunk/ar71xx/generic/packages
/routing/Packages.sig.
Signature check passed.
Downloading http://downloads.openwrt.org/snapshots/trunk/ar71xx/generic/packages
/telephony/Packages.gz.
Updated list of available packages in /var/opkg-lists/chaos_calmer_telephony.
Downloading http://downloads.openwrt.org/snapshots/trunk/ar71xx/generic/packages/telephony/Packages.s
g.
Signature check passed.
root@OpenWrt:~#

```

Kuva 15. Opkg update -komennon suorittaminen

Voimme tarkastella jo asennettuja paketteja opkg list_installed-komennolla ja uusia paketteja, joita voimme asentaa, pääsemme nyt näkemään komennolla opkg list. Komentoja pystyy myös yhdistämään, eli esimerkiksi opkg list | grep samba-komennolla pystymme havaitsemaan kaikki laajennuspaketit, joiden nimessä esiintyy sana samba. [20.]

Laajennuspakettien asentaminen onnistuu opkg update -toiminnon suorittamisen jälkeen komennolla opkg install ja laajennuspaketin nimi tai poikkeustapauksessa opkg --force-depends install ja laajennuspaketin nimi. Tällöin paketti asennetaan niin sanotusti väkisin, eikä sen toimintavarmuudesta voida olla enää varmoja. [20.]

4.4 Samba-tiedostopalvelin

Samba on ilmainen, avoimen lähdekoodin sovellus, joka käyttää SMB-verkkoprotokollaa. Kolmannesta versiosta lähtien Samba pystyy tarjoamaan verkossa tiedosto- ja tulostuspalveluja Windows-, Unix- ja MacOS X -clienteille

Samban OpenWrt-toteutus on joukko asennettavia laajennuspaketteja, joilla voi kätevästi toteuttaa useiden hakemistojen jakamisen lähiverkossa esimerkiksi suoraan reitittimeen liitetyltä, ulkoiselta USB-muistilta. Muita samantyyppisiä vaihtoehtoja ovat OpenWrt:n NFS- ja SSHFS-palvelinlaajennukset.

Samban oli kehittänyt Andrew Tridgell, joka tälläkin hetkellä johtaa Samban kehitystiimiä. Andrew aloitti projektin vuonna 1991 samalla, kun hän työskenteli Pathworks-ohjelmistojen parissa Digital Equipment Corporation -yhtiössä. Pathworksia käytettiin eri valmistajien tietokoneiden yhdistämiseen toisiinsa ja yhdessä vaiheessa Andrew loi tiedostopalvelinohjelmiston yhdelle Pathworksin protokollista, SMB:lle. Vuosia myöhemmin hän alkoi levittämään kehittämänsä SMB-palvelinohjelmistoa, mutta koska nimi oli jo toisen yhtiön käytössä, se vaihdettiin Sambaksi. [21.]

OpenWrt:lle Samba on saatavilla samba36-server laajennuspaketin muodossa. Luci-app-samba on myös hyödyllinen laajennuspaketti, koska se tarjoaa keinon hallita Sambaa OpenWrt:n selainpohjaisen käyttöliittymän Lucin kautta. [22.]

4.4.1 Laajennuspaketit

Jotta voisimme muuttaa OpenWrt-reitittimen lähiverkon tiedostopalvelimeksi tulemme tarvitsemaan kolmea laajennusta: Luci, samba36-server ja luci-app-samba.

Luci koostuu joukosta erilaisia laajennuspaketteja ja tarjoaa meille selainpohjaisen, graafisen käyttöliittymän OpenWrt-reitittimeen. Se ei ole välttämätön, mutta auttaa toiminnallisuuden hahmottamisessa tämän työn kannalta. [23.]

Samba36-server tarjoaa meille Samba-palvelinpaketin, joka mahdollistaa muun muassa tiedosto- ja tulostinpalveluja lähiverkkoon. Näihin palveluihin pääsevät olemaan yhteydessä esimerkiksi Windows, Unix ja MacOS X ympäristöjen clientit. [22.]

Luci-app-samba on laajennus luci-käyttöliittymään, josta voi tarkastella ja säätää Samba-palvelimen asetuksia graafisesti. Seuraavat kernel-moduulit tulee myös asentaa: kmod-scsi-core, kmod-fs-vfat, kmod-usb-storage, kmod-nls-cp437, kmod-nls-iso8859-1. Näiden moduulien avulla enableimme tuen ulkoiselle USB-lisämuistille ja pystymme liittämään sen järjestelmään pysyvästi.

4.4.2 Konfigurointi

Suoritettuaamme `opkg update` -komennon asennamme laajennuspaketit kirjoittamalla `opkg install`. Tämän jälkeen boottaamme laitteen ja asetamme USB-muistitikun reitittimen USB1-porttiin. Kirjaudumme sisään SSH-yhteydellä ja suunnistamme `mnt-hakemistoon`, jonka sisään luomme `mounting point -hakemiston` `mkdir storage` -toiminnolla.

Kirjoittamalla `fdisk -ls` -komennon näemme listan laitteen saatavilla olevista muistipaikoista. Löydämme meidän USB-muistin nimellä `sda1` ja kiinnitämme sen järjestelmään komennolla `mount /dev/sda1 /mnt/storage/folder`. Kun tämä on valmista, käynnistämme laitteen uudelleen ja navigoimme Ubuntu-testikoneen selaimessa `http://192.168.1.1`-sivulle. Kirjaudumme `root/ Qwerty123` -tiedoilla Lucin sisään ja suunnistamme `Services-Network shares` -valikkoon.

The screenshot shows the OpenWrt Luci web interface for configuring Samba. The 'Services' tab is selected, and the 'Network Shares' section is active. Under 'Samba', the 'General settings' are visible. The 'Hostnames' field is set to 'OpenWrt', 'Description' is 'OpenWrt', 'Workgroup' is 'WORKGROUP', and 'Share home-directories' is set to '1'. Below this, the 'Shared Directories' table lists a single share named 'folder' located at the path '/mnt/usb_share/folder'. The table includes columns for 'Name', 'Path', 'Allowed users', 'Read-only', 'Allow guests', 'Create mask', and 'Directory mask'. At the bottom of the interface, there are buttons for 'Reset', 'Save', and 'Save & Apply'.

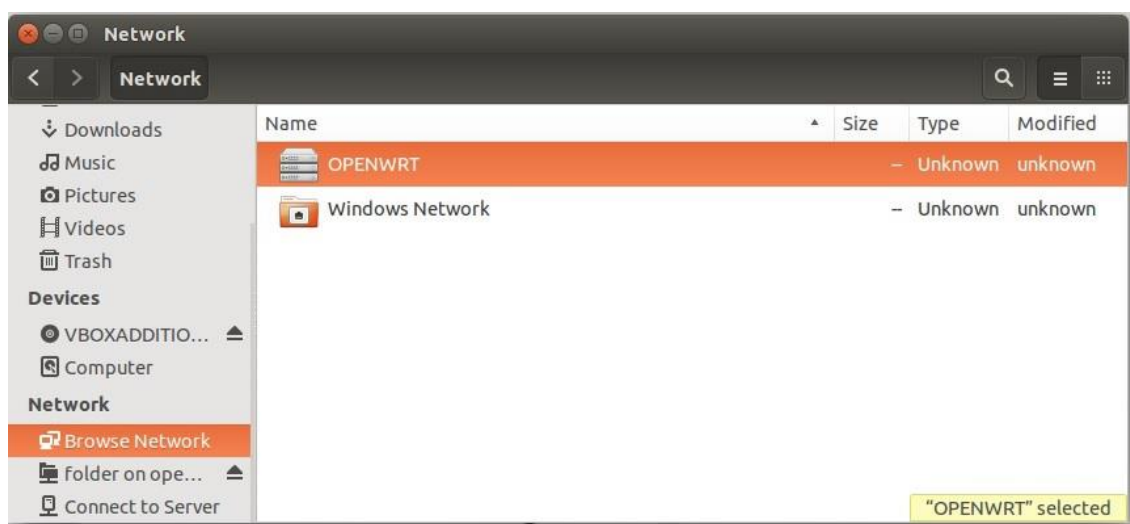
Kuva 16. Tiedostopalvelimen konfigurointi selainpohjaisella käyttöliittymällä

Täältä avaamme Shared Directoriesin ja addin, sekä lisäämme polun /mnt/usb_share/. Tallennamme muutokset ja valitsemme option Allow guests. Workgroupin nimeksi tulee workgroup, ja hostname sekä description ovat OpenWrt. [24.]

Valitsemme Save & Apply:n ja käynnistämme uudelleen laitteen. Kirjaudumme SSH-yhteydellä OpenWrt-reitittimeen ja laskeudumme `cd /mnt/usb_storage/folder-`komennolla folder-nimiseen alihakemistoon. Siellä suoritamme komennon `touch testfile`, joka luo meille tyhjän testitiedoston alihakemistoon.

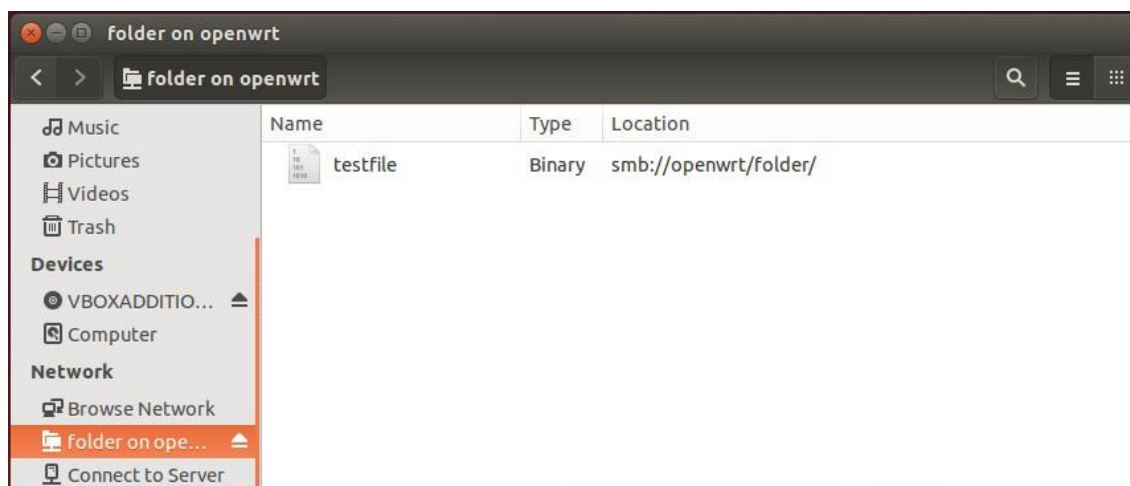
5 Ubuntu client

Ubuntu-testikoneessa avaamme files-valikon ja valitsemme sieltä network. Täällä meille näkyy lähiverkon tiedostopalvelin nimellä OPENWRT.



Kuva 17. Ubuntu client näkee OpenWrt-palvelimen lähiverkossa.

Voimme klikata OPENWRT:n auki, ja alta paljastuu tekemämme jaettu kansio. Sen sisältä löytyy tekemämme testfile. Voimme nyt ladata sen esimerkiksi työpöydälle. Samalla tavalla OpenWrt-palvelin näkyy nyt myös mille tahansa muulle lähiverkon clientille. Tähän jaettuun kansioon voi tallentaa tiedostoja ja sieltä voi myös hakea niitä. Reititimen tasolla tiedostot tallentuvat suoraan sen USB-muistiin.



Kuva 17. Testitiedosto OpenWrt-palvelimen jaetussa hakemistossa

6 Yhteenveto

Työn tavoitteena oli demonstroida yhtä lukuisista OpenWrt-käyttöjärjestelmän sovelluksista käytännön näkökulmasta. Koska tiedostopalvelinkäyttö on yksi selkeimpiä OpenWrt:n ominaisuuksia, se oli suoraviivainen ja helposti ymmärrettävä tapa tarjota konkreettinen esimerkki siitä, kuinka OpenWrt:n laajennettavuus toteutuu.

Opinnäytetyön fokuksena pyrittiin pitämään juuri OpenWrt Buildrootin ja OpenWrt-käyttöjärjestelmän soveltamista. Perimmäinen tavoite, eli lähiverkossa toimivan tiedostopalvelimen rakentaminen OpenWrt-alustalle saavutettiin, ja testitiedoston lataaminen palvelimen ulkoiselta USB-muistilta osoittautui myös onnistuneeksi Ubuntu clientille.

Yksi haasteista työn toteutuksessa oli OpenWrt-yhteisön suhteellinen hajanaisuus, joten neutraalin ja ajankohtaisen tiedon hakeminen OpenWrt:sta ei aina ollut helppoa. Tällä hetkellä OpenWrt:n tulevaisuus oivana työkaluna oppimiseen ja sulautettujen järjestelmien toiminnan ja sovellusten kehittämiseen näyttää ilman muuta valoisalta.

OpenWrt:n onnistumisen puolesta puhuu jo yli kymmenen vuoden elinkaari ja jatkuvasti kasvava yhteisö sen takana.

Lähteet

- [1] OpenWrt build system – Installation [web-sivu], OpenWrt kotisivu [Luettu 7.10.2015]. Luettavissa: <http://wiki.openwrt.org/doc/howto/buildroot.exigence>.
- [2] openwrt [web-sivu], SPI kotisivu [Luettu 8.10.2015]. Luettavissa: <http://www.spi-inc.org/projects/openwrt/>.
- [3] OpenWrt Development List [web-sivu], GMANE kotisivu [Luettu 1.10.2015]. Luettavissa: <http://comments.gmane.org/gmane.comp.embedded.openwrt.devel/32651>.
- [4] OpenWrt - First Login [OpenWrt Wiki] [web-sivu], OpenWrt Wiki kotisivu [Luettu 1.10.2015]. Luettavissa: <http://wiki.openwrt.org/doc/howto/firstlogin>.
- [5] Linksys WRT54G and the GPL [web-sivu], Linux Kernel Mailing List archive kotisivu [Luettu 2.10.2015]. Luettavissa: <https://lkml.org/lkml/2003/6/7/164>.
- [6] OpenWrt Version History [web-sivu], OpenWrt Wiki kotisivu [Luettu 1.10.2015]. Luettavissa: <http://wiki.openwrt.org/about/history>.
- [7] OpenWrt's build system - About [OpenWrt Wiki] [web-sivu], OpenWrt Wiki kotisivu [Luettu 17.9.2015]. Luettavissa: <http://wiki.openwrt.org/about/toolchain>.
- [8] [OpenWrt-Devel] OpenWrt development process [web-sivu], OpenWrt Mailing Lists kotisivu [Luettu 17.9.2015]. Luettavissa: <https://lists.openwrt.org/pipermail/openwrt-devel/2009-January/003542.html>.
- [9] SubmittingPatches/Svn - OpenWrt [web-sivu], OpenWrt development center kotisivu [Luettu 17.9.2015]. Luettavissa: <https://dev.openwrt.org/wiki/SubmittingPatches/Svn>.
- [10] Git - Git Basics [web-sivu], Git kotisivu [Luettu 17.9.2015]. Luettavissa: <https://git-scm.com/book/en/v2/Getting-Started-Git-Basics>.
- [11] GNU General Public License v2.0 - GNU Project - Free Software Foundation [web-sivu], The GNU Operating System and the Free Software Movement kotisivu [Luettu 17.9.2015]. Luettavissa: <https://www.gnu.org/licenses/gpl-2.0.html>.
- [12] License [OpenWrt Wiki] [web-sivu], OpenWrt Wiki kotisivu [Luettu 1.10.2015]. Luettavissa: <http://wiki.openwrt.org/about/license>.
- [13] The GNU General Public License v2 - An Overview, OSS Watch kotisivu [Luettu 1.10.2015]. Luettavissa: <http://oss-watch.ac.uk/resources/gpl>.

[14] What is Oracle VirtualBox? [web-sivu], PeopleSoft Tutorial kotisivu [Luettu 3.10.2015]. Luettavissa: <http://peoplesofttutorial.com/what-is-oracle-virtualbox/>.

[15] Understanding Hardware-Assisted Virtualization [web-sivu], ADMIN magazine kotisivu [Luettu 1.10.2015]. Luettavissa: <http://www.admin-magazine.com/Articles/Hardware-assisted-Virtualization>.

[16] Deciding between virtual machine or dual-boot setup [web-sivu], Ask ubuntu kotisivu [Luettu 3.10.2015]. Luettavissa: <http://askubuntu.com/questions/133845/deciding-between-virtual-machine-or-duabootsetup>.

[17] Installing Ubuntu inside Windows using VirtualBox [web-sivu], psychocats.net kotisivu [Luettu 3.10.2015]. Luettavissa: <http://www.psychocats.net/ubuntu/virtualbox>.

[18] OpenWRT Docs - of /kamikaze [web-sivu], OpenWrt Downloads kotisivu [Luettu 3.10.2015]. Luettavissa: <https://downloads.openwrt.org/kamikaze/docs/openwrt.html>.

[19] TP-LINK TL-WDR4300 - WikiDevi [web-sivu], Wiki Devi kotisivu [Luettu 7.10.2015]. Luettavissa: https://wikidevi.com/wiki/TP-LINK_TL-WDR4300.

[20] OPKG Package Manager [web-sivu], OpenWrt Wiki kotisivu [Luettu 7.10.2015]. Luettavissa: <http://wiki.openwrt.org/doc/techref/opkg>.

[21] Samba: An Introduction [web-sivu], Samba kotisivu [Luettu 7.10.2015]. Luettavissa: <https://www.samba.org/samba/docs/SambaIntro.html>.

[22] Where to get packages [web-sivu], OpenWrt Wiki kotisivu [Luettu 7.10.2015]. Luettavissa: <http://wiki.openwrt.org/doc/packages>.

[23] LuCi Essentials [web-sivu], OpenWrt Wiki kotisivu [Luettu 7.10.2015]. Luettavissa: <http://wiki.openwrt.org/doc/howto/luci.essentials>.

[24] Share USB Hard-drive with Samba using the Luci web-interface [web-sivu], OpenWrt Wiki kotisivu [Luettu 7.10.2015]. Luettavissa: <http://wiki.openwrt.org/doc/recipes/usb-storage-samba-webinterface>.